Apprentissage Statistique

et

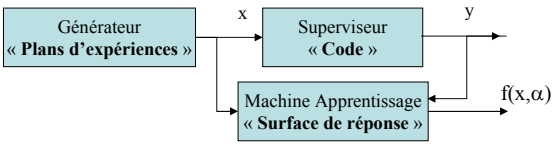
Techniques de Ré-échantillonnage

Jean-Marc.Martinez@cea.fr
Centre d'Etudes de Saclay
Département de Modélisation des Systèmes et Structures
91190 Gif sur Yvette

- Ingénieur-Chercheur au CEA, Commissariat à l' Énergie Atomique, au Centre d'Études de Saclay au DM2S, Département de Modélisation des Systèmes et Structures
- Domaine d'activité : Simulation numérique
  - Traitements d'incertitudes
  - Analyses de sensibilité
  - Supervision des calculs
- Méthodes
  - Plans d'expériences numériques
  - Surfaces de réponse non linéaires
  - Apprentissage statistique (réseaux de neurones)
  - Optimisation multicritère (algorithmes génétiques)

Apprentissage Statistique (Vapnik)

- Modèle général de l'apprentissage à partir d'exemples



  - ***Générateur*** vecteurs aléatoires $x \in R^n$, générés à partir d'une distribution de probabilité F(x)
  - ***Superviseur*** qui associe une valeur y à chaque x, en fonction distribution conditionnelle F(y|x) inconnue
  - ***Machine d'Apprentissage*** opérant sur un ensemble de fonctions paramétriques $f(x,\alpha)$, $\alpha \in \Lambda$ ensemble de paramètres

$\Rightarrow$ La sélection d'une fonction $f(x,\alpha)$ est basée sur l'ensemble d'apprentissage composé des observations $(x_1,y_1),\ldots,(x_p,y_p)$

Évaluation de l'Apprentissage

- Fonction de coût $L(y,f(x,\alpha))$ entre la réponse y du Superviseur et la prédiction $f(x,\alpha)$ fournit par la Machine d'Apprentissage

Coût Moyen= Risque fonctionnel → $R(\alpha) = \int L(y, f(x,\alpha)) dF(x,y)$

- Le ***risque fonctionnel*** $R(\alpha)$ n'est pas calculable; il est remplacé par le ***risque empirique*** calculé à partir des observations

$$R_{\text{emp}}(\alpha) = \frac{1}{N} \sum_i L(y_i, f(x_i,\alpha))$$

  - **Problème** : Le minimum de $R(\alpha)$ peut être différent du minimum de $R_{emp}(\alpha)$ (approximation de fonction = problème ***mal posé***)

- L'objectif de la **Théorie de l'Apprentissage Statistique** est de définir les conditions nécessaires et suffisantes pour assurer :

$$\text{Condition (C)} = \left\{ \text{Arg}_{\alpha}\text{Min } R(\alpha) \approx Arg_{\alpha} Min\ R_{\text{emp}}(\alpha) \right\}$$

Cas de la régression

- Probabilité conditionnelle du Superviseur P(y|x)
  - Cas déterministe ⇒ *certitude conditionnelle* :

$$p(y|x) = \delta(y - f_0(x))$$

  - Cas stochastique ⇒ fonction de régression :

$$f_0(x) = \int y\, p(y|x)\, dy$$

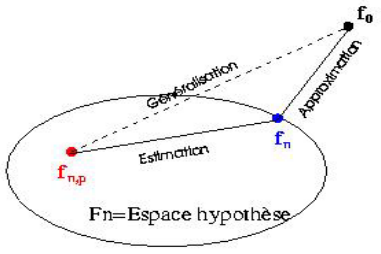
- Décomposition du coût moyen quadratique

$$R(\alpha) = \int [y - f(x,\alpha)]^2\, p(y|x)\, p(x)\, dx dy$$
$$= \underbrace{\int [f_0(x) - f(x,\alpha)]^2\, p(x) dx}_{R_0(\alpha)=\text{Partie à minimiser}} + \underbrace{\int [y - f(x)]^2\, p(y|x)\, p(x) dx dy}_{\text{Variance conditionnelle}}$$

⇒ La fonction de régression $f(x,\alpha)=f_0(x)$ est la solution optimale

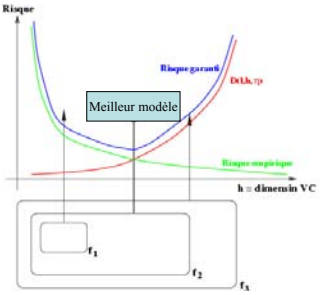
Décomposition du Risque $R_0(\alpha)$

- Erreur d'approximation
  - La solution optimale $f_0(x)$ peut ne pas être représentée par une fonction paramétrique $f_n(x,\alpha)$
  - car $f_0(x) \notin$ *l'espace hypothèse* F_n contenant les fonctions paramétriques :
    - polynômes de degré ≤ n
    - réseaux de neurones à une couche cachée de n unités
- Erreur d'estimation
  - La solution $f_{n,p}(x,\alpha)$ minimisant le risque empirique est différente de la solution optimale du risque fonctionnelle
  - car on ne dispose que d'un nombre limité d'observations



Théorie de Vapnik et Chervonenkis

- La ***dimension*** VC d'un ensemble de fonctions $f_n(x,\alpha)$ mesure leurs ***capacités de modélisation***
  - si $h(f_n)$ dimension VC de $f_n(x,\alpha)$=polynôme degré $\leq n$ alors $h(f_{n+1})>h(f_n)$



- Théorème : ***L'erreur d'estimation est bornée***

$$\text{Prob}[\,|R(\alpha)-R_{emp}(\alpha)|<D(\text{p,h},\eta)\,]>1-\eta$$
$$\Rightarrow R(\alpha)\approx \text{Re}mp(\alpha)+D(p,h,\eta)$$

Intervalle 1-η confiance

- $R_{emp}(\alpha)$ est décroissante mais D(h) est croissante en fonction de la dimension VBC $\Rightarrow$ compromis
- On ne sait pas calculer avec précision la dimension VC pour des fonctions non linéaires
- ***solutions issues des réseaux de neurones et des méthodes statistiques***

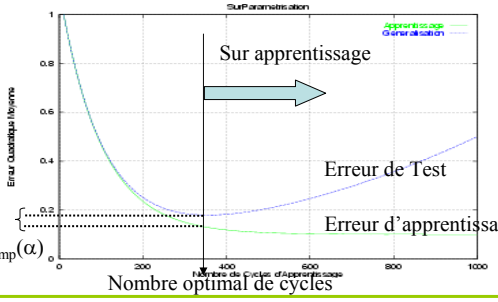
Pourquoi les réseaux de neurones

- Propriétés
  - **Stinchecombe and White 1986** : toute fonction R^n to R^p peut être approchée par un réseau de neurones multicouche
    - Une couche cachée à fonction logistique suffit
  - **Baron 1991** : Pour des réseaux de neurones (p unités en couche cachée), l'erreur d'approximation en O(1/p) ***indépendante de la dimension de l'espace d'entrée***

    $\Rightarrow$ intérêt des réseaux multicouche en grande dimension
  - Dans de nombreuses applications (reconnaissance de formes, régression) leur utilisation a montré de bonnes **capacités en généralisation**
    - L'intervalle de confiance D(p,h,η) est une a fonction faiblement croissante de la dimension VC (régularisation inhérente)

Méthodologie

- Validation croisée (cross validation)
  - Diviser l'ensemble des observations en **base d'Apprentissage et base de Test**
    - erreur d'Apprentissage = risque empirique
    - erreur de Test ≈ risque fonctionnel
- Arrêt prématuré (early stoping)
  - Erreur d'apprentissage décroît; l'erreur de Test a un minimum
  - Le nombre de cycles d'apprentissage est adapté en fonction du minimum de l'erreur de test

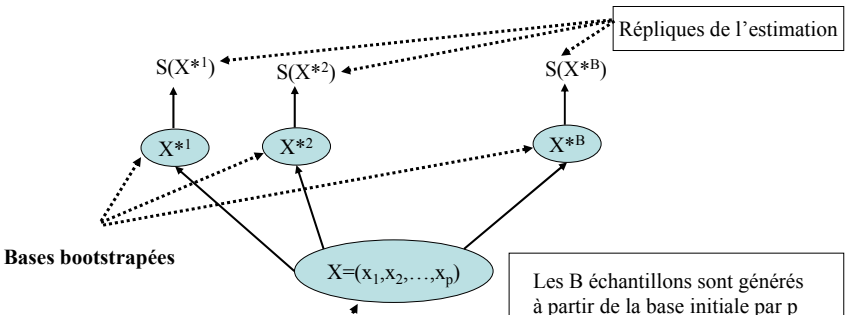


1 cycle = présentation aléatoire de tous les exemples de la base d'apprentissage

Le ***bootstrap*** pour améliorer l'estimation du biais $D(h)=R(\alpha)-R_{emp}(\alpha)$

Bootstrap - Principe

- Bootstrap (Efron 1985) est une technique de rééchantillonnage pour estimer l'écart type et l'intervalle de confiance pour tout estimateur S(X)



Bootstrap - Méthode

- Distribution F inconnue; Θ paramètre de la loi F
  - réalisation d'un échantillon $\mathbf{x}=(x_1,\ldots,x_n)$ distribution $\hat{F}$
  - estimation de Θ par $\theta=S(x)$
  - Comment calculer l'écart type σ de l'estimateur $\theta=S(\mathbf{x})$?
- Méthode
  - sélectionner B répliques $x^{*1}, x^{*2},\ldots,x^{*B}$ obtenues par n tirages aléatoire avec remise
  - pour chaque réplique $\mathbf{x}^{*b}$ calculer $\theta^*(b)=S(\mathbf{x}^{*b})$; b=1,2,…,B
  - Estimer l'écart type $\sigma_{\hat{F}}$ par

$$\theta_{mean} = \frac{1}{B}\sum_{b=1}^{B} S(X^{*b})$$

$$\sigma_{boot}{}^2 = \frac{1}{B-1}\sum_{b=1}^{B}\left(S\left(X^{*b}\right)-\theta_{mean}\right)^2$$

- Théorème

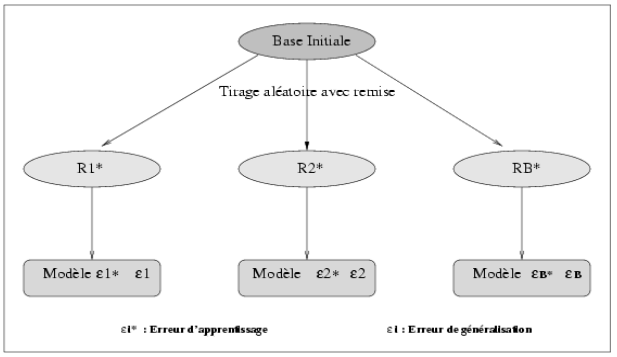
$$\lim_{B\to\infty}\sigma_{boot} = \sigma_{\hat{F}}$$

Bootstrap – Erreur de généralisation

- Le bootstrap permet d'estimer l'écart entre l'erreur d'apprentissage (risque empirique) et l'erreur de généralisation (risque fonctionnelle)
- Apprentissage →bases *bootstrapées*, test → base initiale

L'analyse des erreurs de test en fonction du nombre de cycles permet de déterminer l'arrêt prématuré optimal

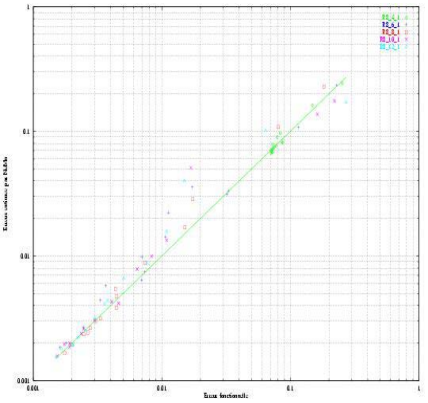
Plusieurs stratégies (fractiles, trimédian) permettent d'exclure les apprentissages «outliers »



Validation de la méthode « *NeMo* »

- Cas Test
  - Fonction $f_0(x)$ de $R^8 \rightarrow R$
  - Apprentissage de la fonction $f_0(x)$ par une famille de réseaux de neurones « 8_x_1 » en faisant varier le nombre x d'unités cachées
  - Nombre d'exemples retenus dans la base : 100, 200, …, 1500
  - Réalisation des apprentissages automatiques avec estimation de l'erreur fonctionnelle
  - Récupération de la fonction analytique et calcul de l'erreur fonctionnelle par Monte Carlo

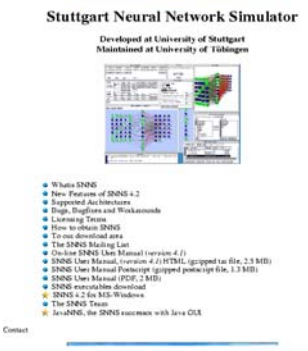
- Comparaison de l'erreur fonctionnelle (abscisse) et de son estimation (ordonnée)



NeMo tool « *Neurones et Modélisation* »

- Logiciel
  - Apprentissage automatique des réseaux de neurones non bouclés (early stopping+ ré-échantillonnage, validation croisée)
  - fournit le programme C du réseau de neurones (fonction + jacobienne)
  - Indicateurs de la modélisation (analyse de l'erreur de généralisation : moments, histogramme; distribution empirique des résidus)
  - utilisable sans connaissance sur les réseaux de neurones
  - s'appuie sur SNNS Stuttgart Neural Network Simulator
    - http://www-ra.informatik.uni-tuebingen.de/SNNS

- Applications
  - Calcul scientifique (neutronique, mécanique, hydrogéologie, …) dans la modélisation de *surfaces réponse non linéaires*



Références

- Livres
  - G. Dreyfus, J.-M. Martinez, … *Réseaux de Neurones – Méthodologie et Applications* – Eyrolles 2002
  - B. Efron, R.J. Tibshirani An introduction to the Bootstrap – Chapman & Hall 1993
  - S. Thiria, Y. Lechevallier, … *Statistique et Méthodes Neuronales* – Dunod 1997
  - Ch. Bishop *Neural Networks for Pattern Recognition* – Oxford University Press 1995
  - V. Vapnik The nature of Statistical Learning Theory – Springer 1995